Amendments to the Claims

1.   (Currently Amended)   A method of automatically invalidating data
2   cached in a cache system <u>physically separate from a data server configured to originate</u>
<u>data to be cached in the cache system,</u> the method comprising:
4   <u>at the cache system,</u> caching a first data item received from <u>the</u> a data server for
service in response to requests to view said first data item;
6   receiving at the cache system a change request to alter a data item; and
automatically invalidating said first data item at the cache system in response to
8   said change request.

2.   (Original)   The method of claim 1, further comprising:
2   comparing said change request to a set of rules for determining when to
automatically invalidate a data item.

3.   (Original)   The method of claim 2, wherein said data item to be altered
2   comprises said first data item.

4.   (Original)   The method of claim 1, wherein said automatically
2   invalidating comprises identifying a relationship between said data item to be altered and
said first data item.

5.   (Original)   The method of claim 1, wherein said cache system
2   comprises multiple caches and said receiving occurs at a first of the multiple caches, the
method further comprising:
4   sending a notification of said automatic invalidation from the first cache of the
multiple caches to a second cache of the multiple caches.

6.   (Currently Amended)   A method of operating a cache system to
2   facilitate automatic invalidation of cached data, <u>the method being performed at the cache</u>
<u>system and</u> comprising:

A1

4   caching a first set of data received from a data server;

receiving at the cache system a first request to change a second set of data;

6   retrieving from said first request an identifier of said second set of data; and

automatically invalidating said first set of data in the cache system without

8   awaiting an invalidation communication from the data server.

7.   (Original)   The method of claim 6, wherein said second set of data

2   comprises said first set of data.

8.   (Original)   The method of claim 6, wherein said second set of data is a

2   later version of said first set of data.

9.   (Original)   The method of claim 6, further comprising identifying a

2   relationship between said second set of data and said first set of data.

10.   (Original)   The method of claim 9, wherein said identifying a

2   relationship comprises:

comparing a pattern of said first request to a first rule for determining when to

4   automatically invalidate a set of data.

11.   (Currently Amended)   The method of claim 10, wherein said first

2   rule comprises a first pattern for identifying a request in response to which a set of data

may be automatically invalidated, and further comprises a second pattern, different from

4   said first pattern, to identify said set of data to be automatically invalidated.

12.   (Original)   The method of claim 6, wherein the cache system

2   comprises multiple caches and:

wherein said caching comprises caching said first set of data at a first cache of the

4   multiple caches; and

wherein said automatically invalidating is performed at a second cache of the

6   multiple caches;

the method further comprising notifying the first cache, by the second cache, of

8   said automatic invalidation.

13.   (Original)   A method of automatically invalidating cached data,

2   comprising:

caching a first set of data at a caching system for serving in response to a view

4   request, wherein said view request comprises a request to view said first set of data, and

wherein said first set of data is received from a data server;

6   receiving at the caching system a change request, wherein said change request

comprises a request to change said first set of data;

8   identifying said first set of data from said change request; and

automatically invalidating said cached first set of data without waiting for the data

10   server to implement said change request.

14.   (Currently Amended)   The method of claim 13, wherein said data

2   server is ~~not~~ notified of said change request only ~~until~~ after said automatic invalidation of

said cached first set of data.

15.   (Currently Amended)   The method of claim 13, wherein said first

2   set of data comprises a <u>first user's bid on</u> ~~price of~~ an item being auctioned in an electronic

auction, and wherein said change request comprises a <u>second user's</u> ~~new~~ bid on said item.

16.   (Original)   The method of claim 13, wherein said first set of data

2   comprises information concerning one or more products at a publicly accessible network

location, and wherein said change request comprises a change to said list of products.

17.   (Original)   The method of claim 13, wherein said receiving a change

2   request comprises receiving a sequence of communications matching a predetermined

pattern.

18.   (Currently Amended)   A computer readable storage medium

2   storing instructions that, when executed by a computing device, cause the computing

device to perform a method of operating a cache system to facilitate automatic

4   invalidation of cached data, the method <u>being performed at the cache system and</u>

comprising:

6   caching a first set of data received from a data server;

receiving at the cache system a first request to change a second set of data;

8   retrieving from said first request an identifier of said second set of data; and

automatically invalidating said first set of data in the cache system without
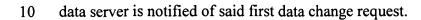
10   awaiting an invalidation communication from the data server.

      19.   (Original)   A computer readable storage medium storing instructions

2   that, when executed by a computing device, cause the computing device to perform a

method of automatically invalidating cached data, the method comprising:

4   caching a first set of data at a caching system for serving in response to a view

request, wherein said view request comprises a request to view said first set of data, and

6   wherein said first set of data is received from a data server;

receiving at the caching system a change request, wherein said change request

8   comprises a request to change said first set of data;

identifying said first set of data from said change request; and

10   automatically invalidating said cached first set of data without waiting for the data

server to implement said change request.

      20.   (Currently Amended)   A cache system configured to automatically

2   invalidate cached data, comprising:

a first cache configured to cache data received from a data server<u>, wherein the</u>

4   <u>data server is coupled to the cache system via a network link</u>;

a data service module configured to serve a first set of cached data in response to

6   a first data view request from a client; and

an invalidation module configured to automatically invalidate said first set of

8   cached data when a first data change request is received from a client;

wherein said automatic invalidation is performed at the cache system before the

10   data server is notified of said first data change request.

21.   (Original)   The cache system of claim 20, wherein said first cache

2   comprises said invalidation module.

22.   (Original)   The cache system of claim 20, further comprising a set of

2   rules for determining when said first set of cached data is to be automatically invalidated

in response to a data change request.

23.   (Original)   The cache system of claim 22, further comprising a user

2   interface configured to facilitate the creation of one of said rules.

24.   (Original)   The cache system of claim 20, wherein said first data

2   change request comprises a predetermined sequence of communications.

25.   (Original)   The cache system of claim 20, further comprising:

2   a second cache;

wherein said second cache is notified by said first cache of said automatic

4   invalidation.

26.   (Withdrawn)

27.   (Withdrawn)